Linear Regression with Classification

About the Dataset:

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by the British statistician and biologist Ronald Fisher in his 1936 paper "The use of multiple measurements in taxonomic problems" as an example of linear discriminant analysis. This famous iris data set gives the measurements in centimeters of the variables sepal length and width and petal length and width, respectively, for 50 flowers from each of 3 species of iris. The species are Iris setosa, versicolor, and virginica.

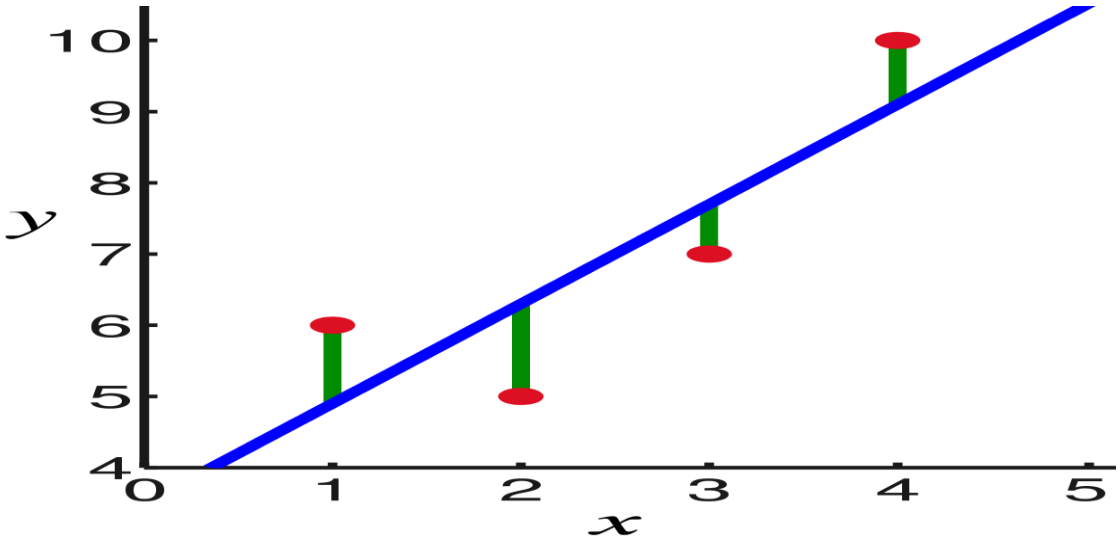
The dataset contains a set of 150 records under 5 attributes -

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. Species: -- Iris Setosa -- Iris Versicolour -- Iris Virginica

Method:

The **linear regression** is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables).

Linear Regression is the concept of drawing the line such that distance from all the points is the lowest from the line.



Linear Regression Equation:

$$Y_i = \beta_0 + \beta_1 X_i + \epsilon_i \quad \text{where} \quad \epsilon_i \sim^{iid} N(0, \sigma^2)$$

Consider now writing an equation for each observation:

$$\begin{aligned} Y_1 &= \beta_0 + \beta_1 X_1 + \epsilon_1 \\ Y_2 &= \beta_0 + \beta_1 X_2 + \epsilon_2 \\ &\vdots \quad \vdots \quad \vdots \\ Y_n &= \beta_0 + \beta_1 X_n + \epsilon_n \end{aligned}$$

The SLR Model in Matrix Form

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} \beta_0 + \beta_1 X_1 \\ \beta_0 + \beta_1 X_2 \\ \vdots \\ \beta_0 + \beta_1 X_n \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} = \begin{bmatrix} 1 & X_1 \\ 1 & X_2 \\ \vdots & \vdots \\ 1 & X_n \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

$$\mathbf{Y}_{n\times 1} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix} \qquad \mathbf{X} = \begin{bmatrix} 1 & X_{1,1} & X_{1,2} & \cdots & X_{1,p-1} \\ 1 & X_{2,1} & X_{2,2} & \cdots & X_{2,p-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_{n,1} & X_{n,2} & \cdots & X_{n,p-1} \end{bmatrix}$$

Vector of Parameters

$$\beta_{2\times 1} = \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}$$

Vector of Error Terms

$$\epsilon_{n\times 1} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}$$

Vector of Responses

$$\mathbf{Y}_{n\times 1} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_n \end{bmatrix}$$

Thus,

$$\begin{aligned} \mathbf{Y} &= \mathbf{X}\beta + \epsilon \\ \mathbf{Y}_{n\times 1} &= \mathbf{X}_{n\times 2}\beta_{2\times 1} + \epsilon_{n\times 1} \end{aligned}$$

I have used Y = Xβ + ε for the linear regression and then used cross validation for finding the accuracy and the error of the data for each fold of the validation.

Cross-validation:

K-Fold Cross-Validation

Primary method for estimating a tuning parameter λ (such as subset size)

- Divide the data into K roughly equal parts

1	2	3	4	5
Validation	Train	Train	Train	Train

- for each $k = 1, 2, \ldots K$, fit the model with parameter λ to the other $K - 1$ parts, giving $\hat{\beta}^{-k}(\lambda)$ and compute its error in predicting the kth part:
  $E_k(\lambda) = \sum_{i \in kth\ part}(y_i - \mathbf{x}_i\hat{\beta}^{-k}(\lambda))^2$.

The concept of K-fold cross validation is that we divide the data into small chunks with the k value and then we train the one part and test on other part. For Ex: n=150(dataset) and k=5, then each chunk is the size of the 30. As shown above, we test on the first chunk and train other 4 chunks. For the next iteration, we test second chunk and train the other 4 chunks and so on.

Result:

I have tested the dataset with different values for the k for cross validation and found the following results.

1) When k=2, I got accuracy as 49.33336% and error is 50.66666%
2) When k=3, I got accuracy as 35.33336% and error is 64.66666%
3) When k=5, I got accuracy as 92.66667% and error is 7.33333%
4) When k=10, I got accuracy as 95.33334% and error is 4.66666%

Conclusion:

I have implemented linear regression, trained the dataset, tested the dataset, used cross validation to find the accuracy of the model.

As per my analysis, the value of the K(cross validation value) is directly proportional to the accuracy of the model. As K grows more number of chunks are trained so the accuracy of the tested data chunk grows. So I think, ideal value for the K=5, as it starts to show descent accuracy for the testing data chunk.

References:

1) https://en.wikipedia.org/wiki/Iris_flower_data_set
2) http://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data
3) https://en.wikipedia.org/wiki/Linear_regression
4) http://statweb.stanford.edu/~tibs/sta306bfiles/cvwrong.pdf
5) https://www.stat.purdue.edu/~boli/stat512/lectures/topic3.pdf
6) https://stackoverflow.com